

# Применение нейронных сетей к распознаванию символов

Dong Xiao Ni

Школа Сейденберг, Центр стратегических и международных исследований, Университет  
Пейса, Уайт-Плейнс, Нью-Йорк

## Аннотация

В этой статье описывается основной биологический нейрон и искусственная вычислительная модель; описывается сеть архитектуры и процессы обучения; и представляются многослойные сети прямой связи. Также представляется два OCR демо-приложений, один с VB.Net, один с C # .Net

## Введение

Искусственная нейронная сеть (ИНС) - это парадигма обработки информации, которая вдохновляется биологическими нервными системами, такие как мозг, информация о процессе. Ключевой элемент парадигмы ИНС является новой структурой системы обработки информации. Он состоит из большого количества взаимосвязанных элементов обработки (нейроны), работающие в унисон для решения конкретных задач. ИНС, например, как люди, учиться на примере. ИНС настроен для конкретного приложения, например распознавания образов или классификации данных через процесс обучения. Обучение в биологических системах включает в себя корректировки синаптических связей, существующие между нейронами. Это относится и к ИНС.

Нейронная сеть представляет собой мощный инструмент моделирования данных, который способен захватывать и представлять сложные отношения ввода / вывода. Мотивация развития технологии нейронных сетей обусловлена стремлением разработать искусственную систему, которая могла бы выполнять «интеллектуальные» задачи, подобные тем, которые выполняются человеческим мозгом. Нейронные сети напоминают человеческий мозг следующими двумя способами: они приобретают знания через обучение, а знания хранятся в пределах межнейронных силовых связей, известных как синаптические веса. Истинная сила и преимущество нейронных сетей заключается в их способности представлять, как линейные, так и нелинейные отношения и в их способности изучать эти отношения непосредственно из моделируемых данных. Традиционные линейные модели просто некомпетентны, когда речь идет о данных моделирования, которые содержат нелинейные характеристики.

Наиболее распространенная модель нейронной сети известна как контролируемая сеть, потому что для ее изучения требуется желаемый результат. Цель этого типа сети - создать модель, которая отображает входные данные на выходе с использованием исторических данных, чтобы затем модель могла использоваться для вывода результата, когда желаемый результат неизвестен. Графическое представление многослойного перцептрона показано на рисунке 1.

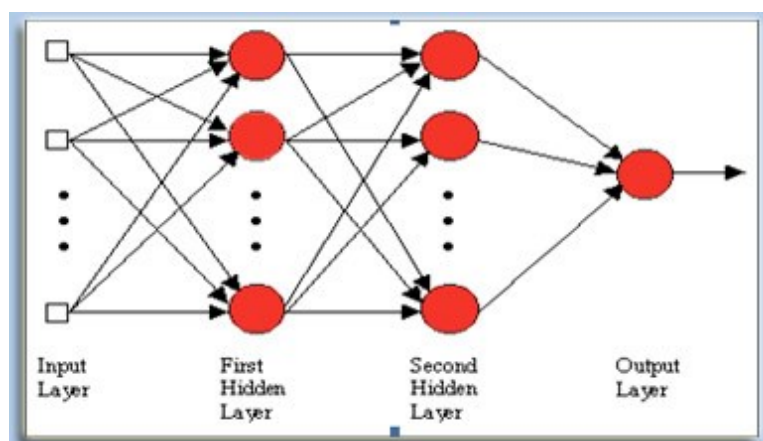


Рисунок 1 - Диаграмма 4-слойного перцептрона с двумя скрытыми слоями

Входы подаются на входной уровень и умножаются на веса межсоединений, поскольку они передаются от входного уровня до первого скрытого уровня. Внутри первого скрытого слоя они суммируются, а затем обрабатываются нелинейной функцией (обычно гиперболическим касанием). Поскольку обработанные данные покидают первый скрытый слой, он снова умножается на веса

межсоединений, затем суммируется и обрабатывается вторым скрытым слоем. Наконец, данные умножаются на весах межсоединений, а затем обрабатываются в последний раз в пределах выходного уровня для получения выхода нейронной сети.

Нейронные сети с их замечательной способностью извлекать смысл из сложных или неточных данных могут использоваться для извлечения шаблонов и выявления тенденций, которые слишком сложны, чтобы их заметили люди или другие компьютерные методы. Обученную нейронную сеть можно рассматривать как «эксперта» в категории информации, которую она давала для анализа. Затем этот эксперт может быть использован для предоставления прогнозов с учетом новых интересующих ситуаций и ответов на вопросы «что, если». Другие преимущества:

- Адаптивное обучение: способность учиться выполнять задания на основе данных, данных для обучения или начального опыта.
- Самоорганизация: ИНС может создать свою собственную организацию или представление информации, которую она получает во время обучения.
- Операция в режиме реального времени: вычисления ИНС могут выполняться параллельно, и разрабатываются и производятся специальными аппаратными устройствами, которые используют эту возможность.
- Отказоустойчивость с помощью избыточного информационного кодирования. Частичное разрушение сети приводит к соответствующему снижению производительности. Однако некоторые сетевые возможности могут сохраняться даже при большом сетевом повреждении.

## Распознавание символов с помощью нейронных сетей

Программы оптического распознавания символов (OCR) способны читать напечатанный текст. Это может быть текст, который был отсканирован из документа, или ручной текст, который был нарисован на ручном устройстве, например, Personal Digital Assistant (PDA). Программы OCR широко используются во многих отраслях. Многие из современных сканеров документов для ПК поставляются с программным обеспечением OCR, которое позволяет сканировать печатный документ, а затем преобразовывать сканированное изображение в электронный текстовый формат, такой как документ Word, позволяя вам манипулировать текстом. Чтобы выполнить это преобразование, программное обеспечение должно анализировать каждую группу пикселей (0 и 1), которые образуют букву, и вырабатывать значение, соответствующее этой букве. Некоторые из программного обеспечения OCR на рынке используют нейронную сеть в качестве механизма классификации.

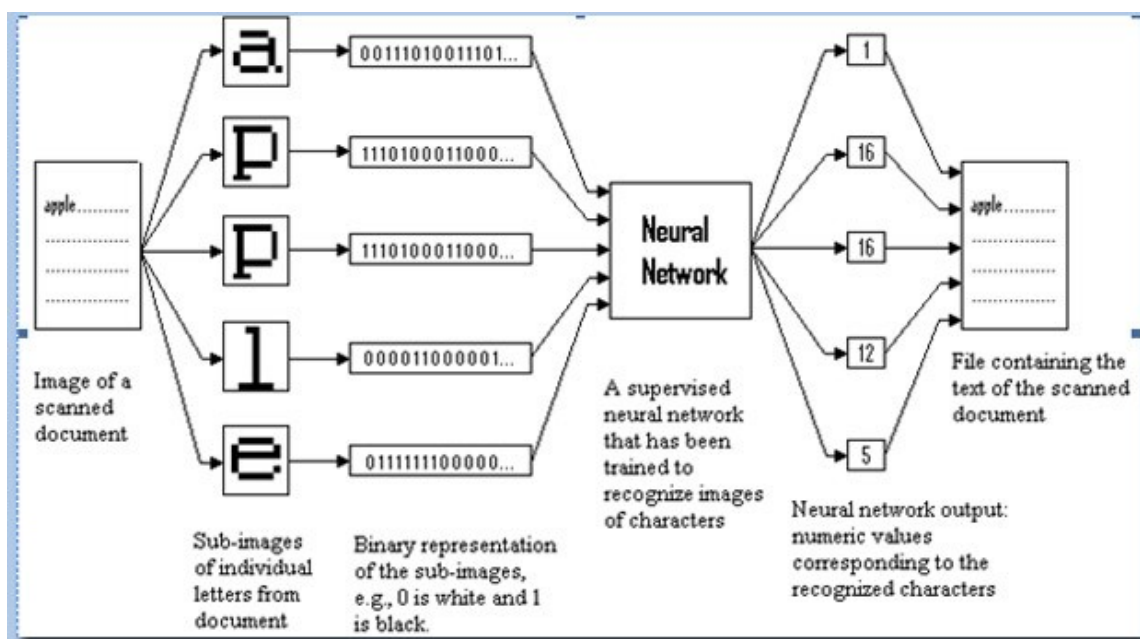


Рисунок 2 - Пример нейронной сети для OCR

На рисунке 2 показана нейронная сеть, используемая в приложении оптического распознавания символов (OCR). Оригинальный документ сканируется в компьютер и сохраняется как изображение. Программное обеспечение OCR разбивает изображение на суб-изображения, каждый из которых содержит один символ. Суб-изображения затем переводятся из формата изображения в двоичный формат, где каждый



Ниже приведены некоторые примеры шаблонов ввода и итоговые результаты нейронной сети. Большинство из результатов довольно хороши.

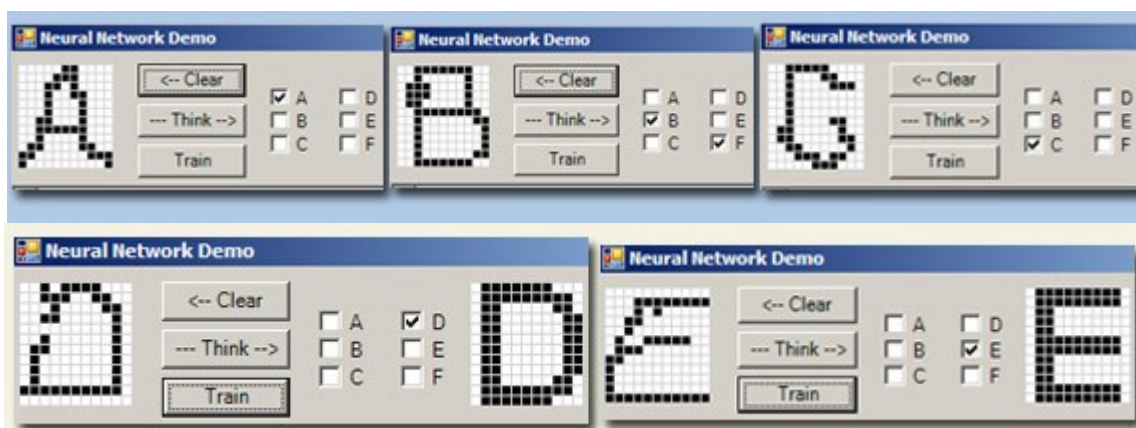


Рисунок 4 - Примеры случаев ввода и довольно хорошие догадки нейронной сети.

И теперь для некоторых тщательно подобранных вариантов, которые должны помочь проиллюстрировать ограничения этого решения.

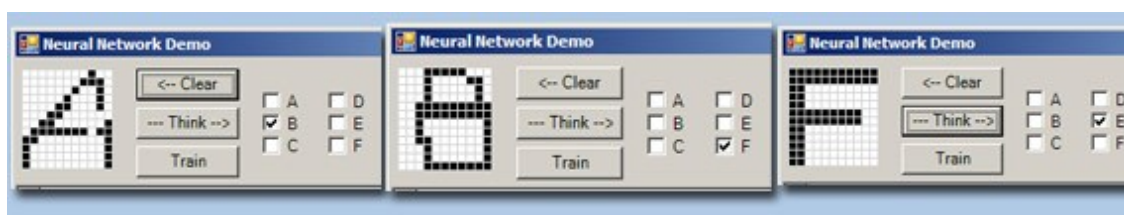


Рисунок 5 - Примеры случаев ввода и довольно плохие догадки нейронной сети.

## Другая нейронная сеть OCR на C#

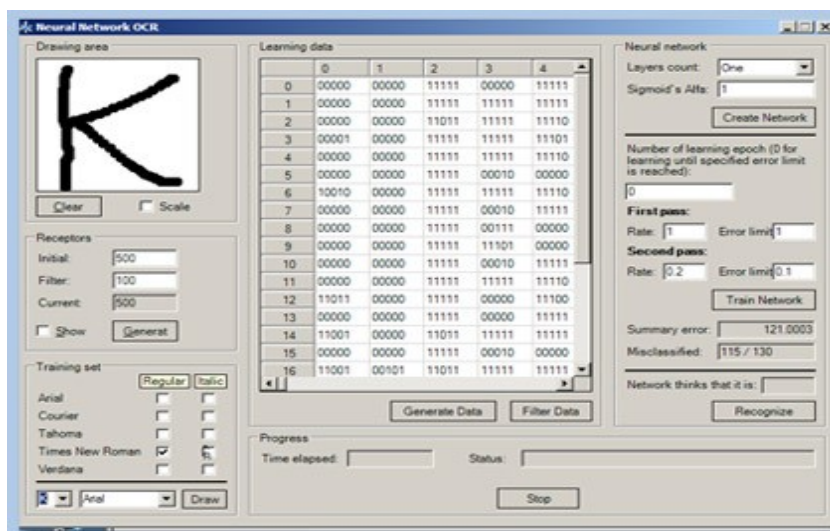
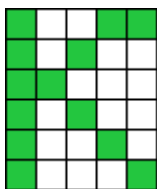


Рисунок 6 - Другое приложение OCR для нейронной сети

Популярный и простой подход к проблеме OCR основан на питательной нервной сети с обратным распространением. Основная идея заключается в том, что учебный набор должен быть подготовлен, а затем обучен нейронной сети, чтобы распознавать шаблоны из учебного набора. На этапе обучения обученная сеть отвечает желаемым выходом для заданного ввода. Для этого каждый образец обучения представлен двумя компонентами: возможным вводом и желаемым выходом сети для ввода. После того, как шаг обучения будет

выполнен, введите произвольный вход в сеть, и сеть сформирует выход, из которого можно разрешить тип шаблона, представленный в сети.



Предположим, что  $t$  для обучения сети распознавать 26 заглавных букв, представленных в виде изображений размером 5x6 пикселей: один из наиболее очевидных способов преобразования изображения во входную часть учебного образца - создать вектор (размером 30 для нашего случая), содержащий «1» во всех положениях, соответствующих буквенному пикселю, и «0» во всех положениях, соответствующих фоновым пикселям. Но во многих задачах обучения нейронной сети предпочтительнее представлять обучающие шаблоны так называемым «биполярным» способом, помещая во входной вектор «0,5» вместо «1» и «-0,5» вместо «0». Такое кодирование шаблонов приведет к повышению эффективности обучения.

Для каждого возможного ввода необходимо создать требуемый выход сети для завершения учебных образцов. Для задачи OCR очень часто кодируется каждый шаблон как вектор размером 26 (потому что есть 26 разных букв), помещая в вектор «0,5» для позиций, соответствующих типу номера шаблона, и «-0,5» для всех других позиций.

### Тестирование приложения

1. Сгенерировать исходные рецепторы. При запуске приложения они уже сгенерированы, поэтому приложение может пропустить этот шаг, если не планирует изменять начальное количество рецепторов или отфильтрованную сумму.

2. Выберите шрифты, которые будут использоваться для обучения сети. Пусть это будет обычный шрифт Arial в первый раз.

3. Сгенерировать данные. На этом этапе будут сформированы исходные данные обучения.

4. Отфильтруйте данные. На этом этапе будут отфильтрованы исходные рецепторы, а также данные обучения.

5. Создать сеть - будет создана нейронная сеть.

6. Тренировка сети - обучение нейронных сетей.

### Вывод

Очевидно, что термин искусственные нейронные сети охватывают большое количество различных программных пакетов со многими различными типами искусственных нейронов, сетевых архитектур и правил обучения. В свою очередь, эти различные сети могут применяться к разнообразным функциям во всем: от производства пива до лучшего понимания свойств биологических мозгов, на которых они основаны

### Список использованной литературы

- [1] José C. Principe, Neil R. Euliano, Curt W. Lefebvre "Neural and Adaptive Systems: Fundamentals Through Simulations", ISBN 0-471-35167-9
- [2] Microsoft web site: Microsoft web site: <http://msdn2.microsoft.com/en-us/default.aspx>
- [3] Christopher M. Frenz, Touch All the Bases: Give Your .NET App Brains and Brawn with the Intelligence of Neural Networks, <http://msdn.microsoft.com/msdnmag/issues/05/05/NeuralNetworks/#S2>
- [4] Articles - Design Patterns, Neural Networks, C#, Programming [http://amazedsaint-articles.blogspot.com/2006/05/brainnet-neural-network-library-part-i\\_22.html](http://amazedsaint-articles.blogspot.com/2006/05/brainnet-neural-network-library-part-i_22.html)
- [5] M. Villani, M. Curtin, Ngo J Simone, H. St. Fort, C. Tappert. S. -H Cha. Keystroke Biometric Recognition Studies on Long Text Input over the Internet, CSIS Pace University, Pleasantville, New York 10570
- [6] Microsoft web site: SQL Server 2005 Books Online "Microsoft Neural Network Algorithm (SSAS)", <http://msdn2.microsoft.com/en-us/library/ms174941.aspx>
- [7] Kingsley Tagbo, C# Neural Network Multilayer Feedforward Backpropagation Algorithm Version 1.0- Open source code release, <http://www.kdkeys.net/blogs/kingsleytagbo/archive/2005/04/09/4514.aspx>, 04/09/2005
- [8] S. Marsland, J. Shapiro, U. Nehmzow. A self-organising network that grows when required. Neural Networks,
- [9] US Postal Service@, Engineering Systems/Process Integration AFCS-OCR, <http://blue.usps.gov/site/wcm/connect/resources/file/eb1591074c8f1ab/SPI%20AFCS-OCR%20Mail%20Flow%20Guide%20v1.1%20Sec%202%20System%20Overview.pdf?MOD=AJPERES>
- [10] Jim Carnicelli, Neural Network Demo, [http://utopia.csis.pace.edu/cs615/2006-2007/team7/Software/OCR\\_vbnet.zip](http://utopia.csis.pace.edu/cs615/2006-2007/team7/Software/OCR_vbnet.zip)
- [11] Andrew Kirillov, Neural Networks, [http://utopia.csis.pace.edu/cs615/2006-2007/team7/Software/OCR\\_Csharp.zip](http://utopia.csis.pace.edu/cs615/2006-2007/team7/Software/OCR_Csharp.zip)
- [12] NeuroIntelligence-Alyuda Research, <http://www.alyuda.com/neural-network-software.htm>
- [13] NeuroDimension Inc web site. - Neural Network Software, <http://www.nd.com/>
- [14] Learn How to Build A Provider Framework For .NET, Articles - Design Patterns, Neural Networks, C#, Programming, <http://amazedsaint-articles.blogspot.com>, March 15, 2007